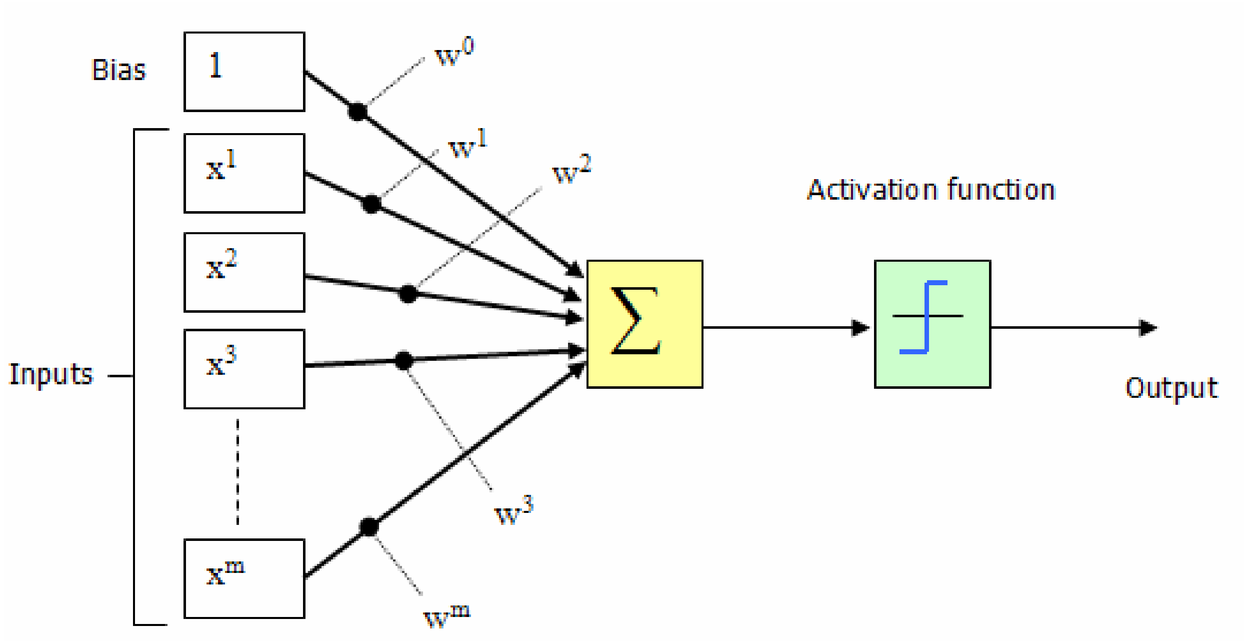
University of Toronto (Mississauga Campus)

CSC411- Machine Learning and Data Mining– Neural Network

Tutorial 3 – Feb 2nd , 2007

Single layer network (Perceptron)

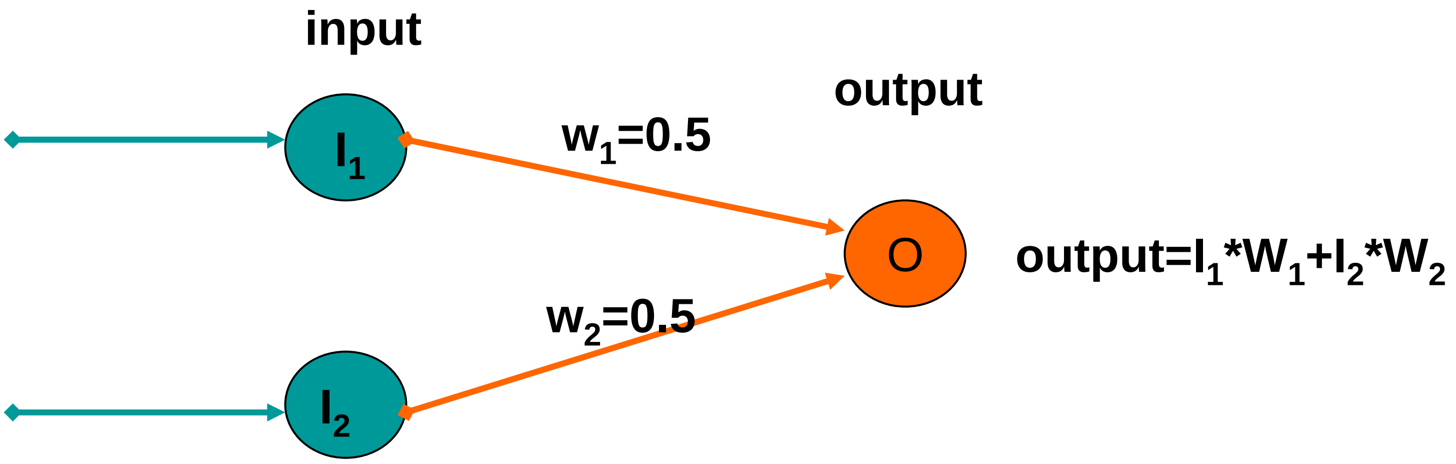


$$\sum_{i=1}^{m} bias + (w^{i} x^{i})$$

Refer to http://www.codeproject.com/useritems/Backprop_ANN.asp

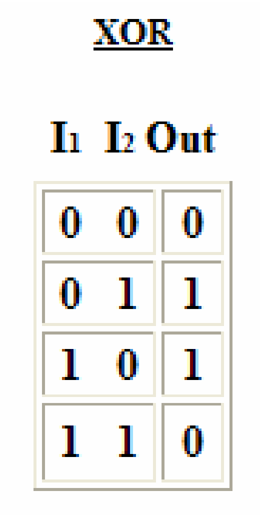
AND Problem:

I_1	I_2	Out
0	0	0
0	1	0
1	0	0
1	1	1



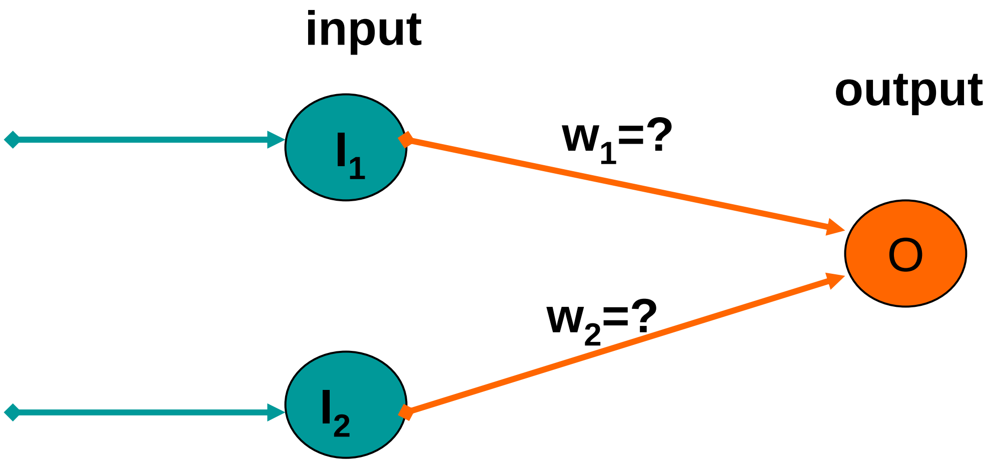
output=I_1*W_1+I_2*W_2

The Classic XOR Problem:



XOR

I_1	I_2	Out
0	0	0
0	1	1
1	0	1
1	1	0



input

output

I_1

I_2

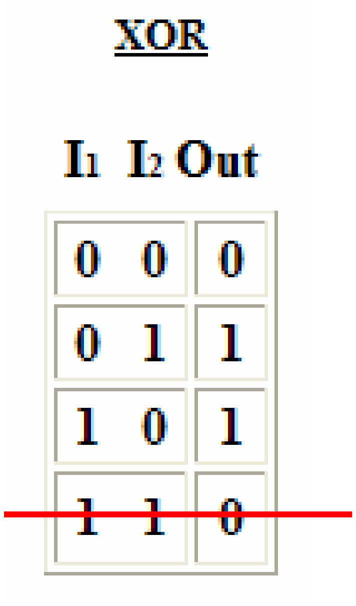
w_1=?

w_2=?

O

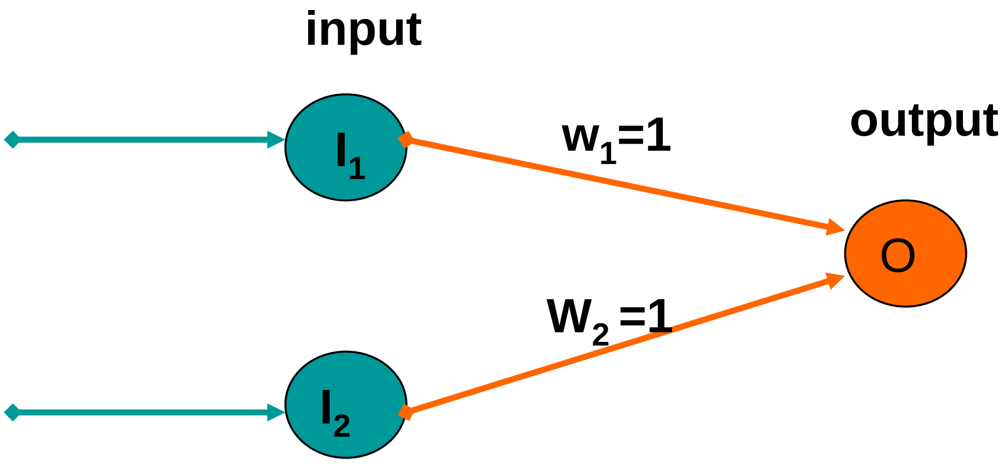
output=I_1*W_1+I_2*W_2

The Classic XOR Problem:



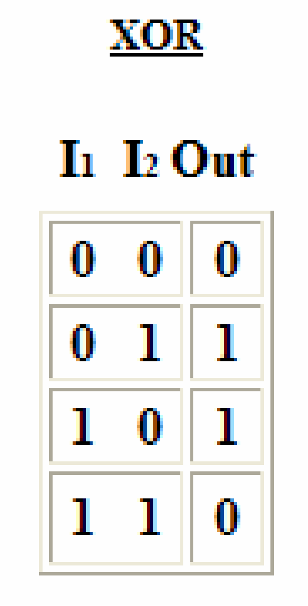
XOR

I_1	I_2	Out
0	0	0
0	1	1
1	0	1
~~1~~	~~1~~	~~0~~



$output=I_1*W_1+I_2*W_2$

The Classic XOR Problem:

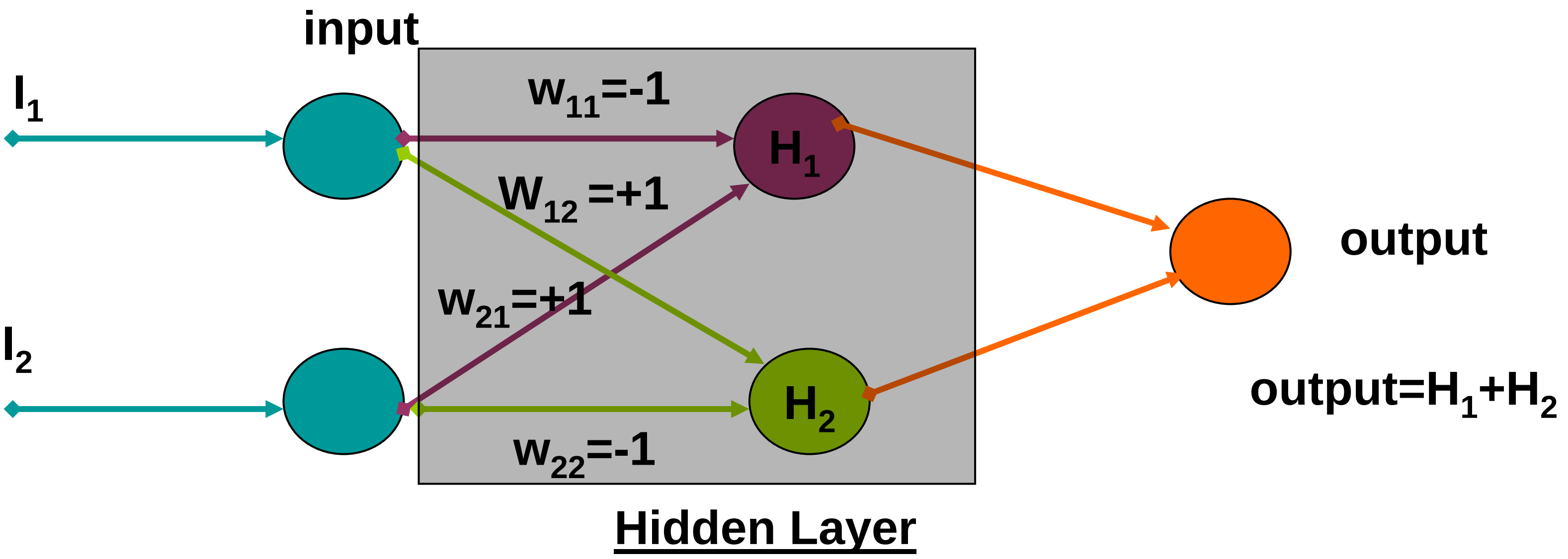


XOR

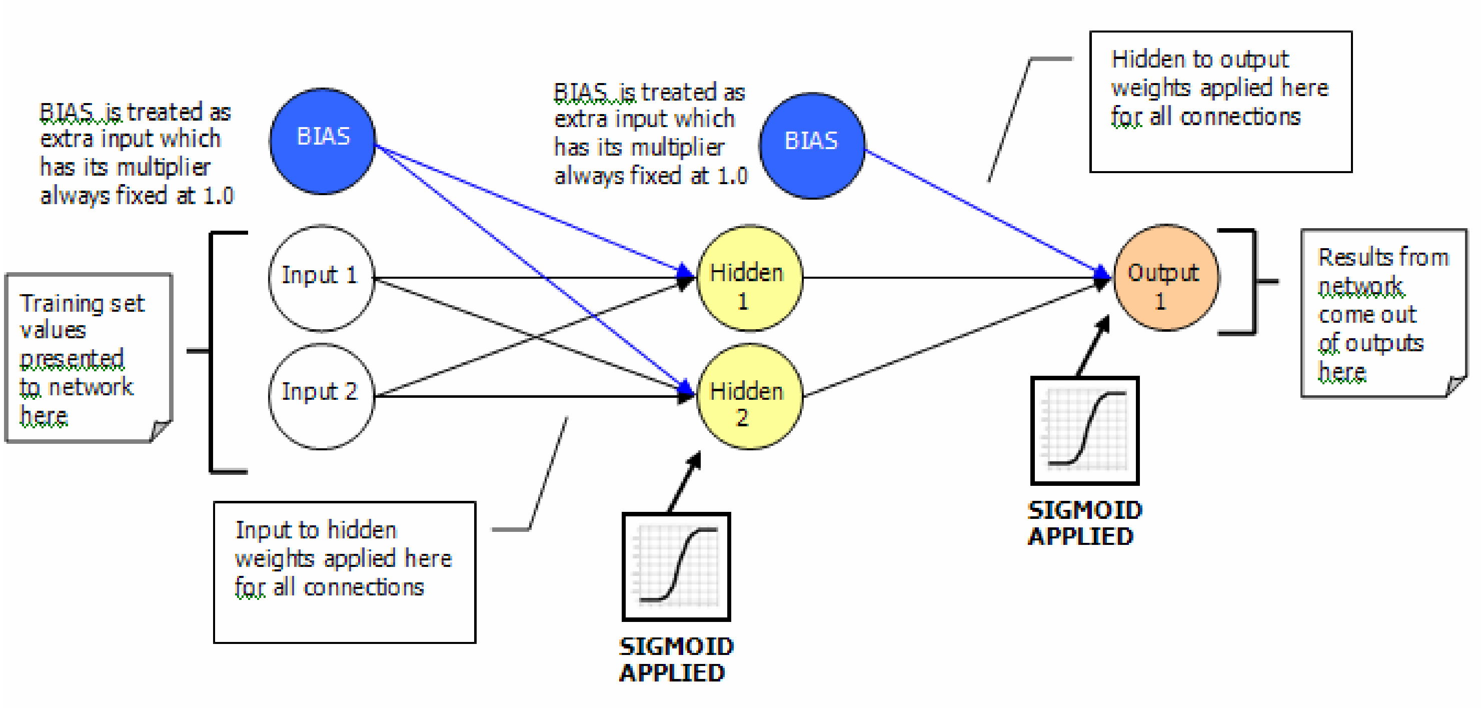
I_1	I_2	Out
0	0	0
0	1	1
1	0	1
1	1	0

$$H_1 = I_1 \times w_{11} + I_2 \times w_{21}$$

$$H_2 = I_2 \times w_{12} + I_2 \times w_{22}$$



Multiple layers



BIAS is treated as extra input which has its multiplier always fixed at 1.0

Training set values presented to network here

Input to hidden weights applied here for all connections

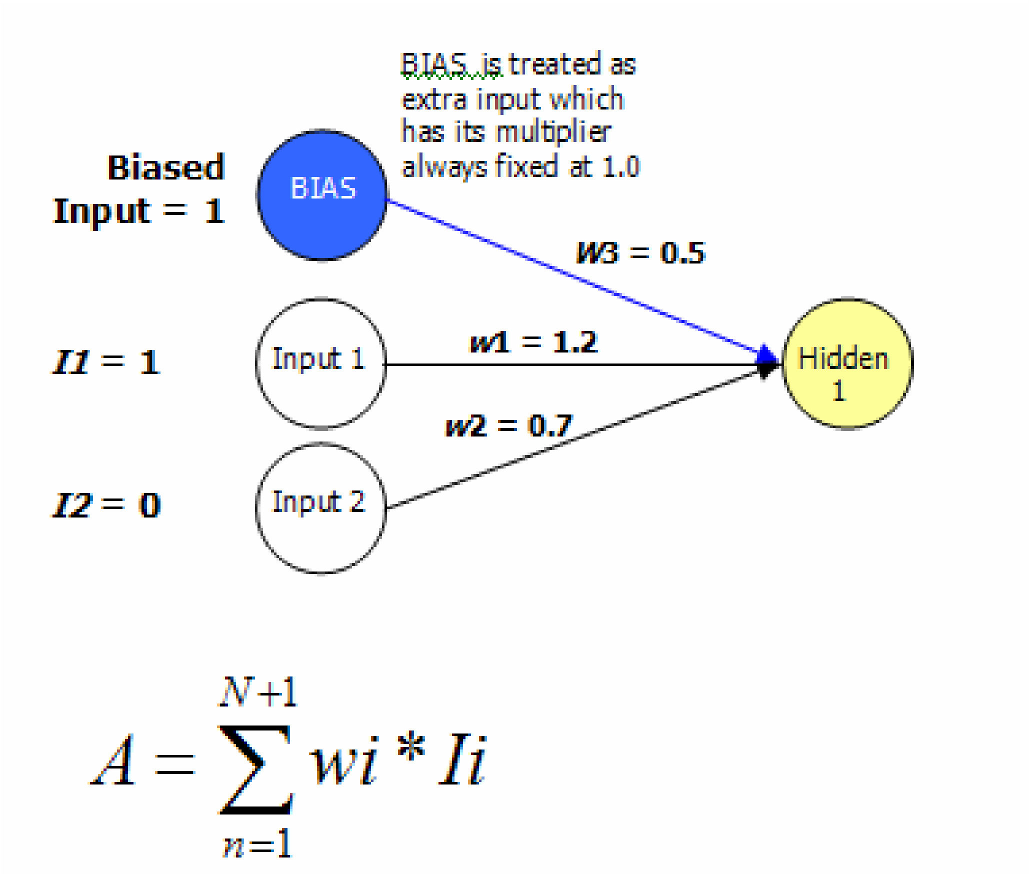
BIAS is treated as extra input which has its multiplier always fixed at 1.0

Hidden to output weights applied here for all connections

Results from network come out of outputs here

Refer to http://www.codeproject.com/useritems/Backprop_ANN.asp

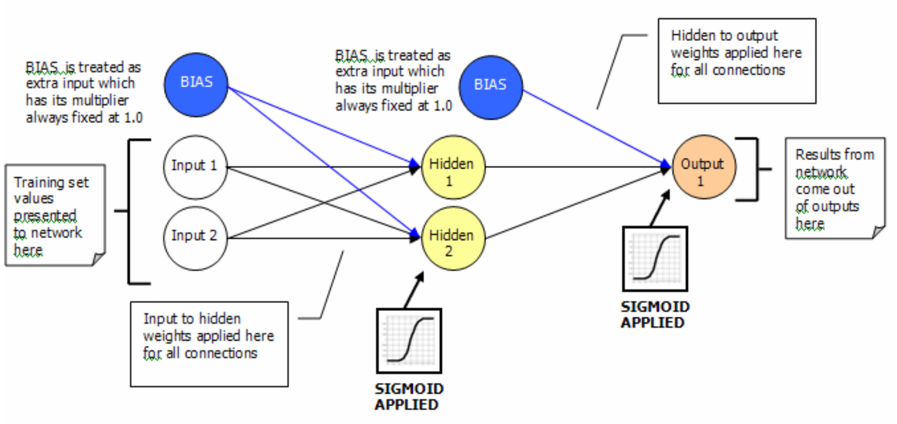
Multiple layers



$$A = \sum_{n=1}^{N+1} wi * Ii$$

Refer to http://www.codeproject.com/useritems/Backprop_ANN.asp

Solve the XOR problem using Back Propagation algorithm



BIAS is treated as extra input which has its multiplier always fixed at 1.0

Training set values presented to network here

BIAS is treated as extra input which has its multiplier always fixed at 1.0

Input to hidden weights applied here for all connections

Hidden to output weights applied here for all connections

Results from network come out of outputs here

$$y = g(x) = \frac{1}{1+e^{-x}}$$

$$\frac{dg}{dx} = g'(x) = g(x)(1 - g(x))$$

```
delta_outputs[i] = outputs[i] * (1.0 - outputs[i]) * (targets[i] - outputs[i])
```

Refer to http://www.codeproject.com/useritems/Backprop_ANN.asp